

HistVis: Making the web browser history useful

Paulo Ferreira

Instituto Superior Técnico
Av. Rovisco Pais, 1, 1049-001
Lisbon, Portugal
Paulo.ferreira55@gmail.com

ABSTRACT

The internet browsed history is a revisitation tool for websites that is a part of the web browsers. However this tool has some problems associated with its use by the users, as for example to be just a textual information, does not contain much information that gives a context for the user and it's separated from the main user interface in the web browsers. The goal that we want in this project is to create a more effective way of using the history resorting to visualization techniques helping the user to recognize and find again the previously visited websites. For this we created a history that represents graphically all the visits done by the user to websites, together with several features that allow to facilitate the search and information visualization. In this scope the user can filter his browsed history in a customized way, that also contains various scalability techniques. A user evaluation shows that the system implemented by us is easily accepted and noticeable, with a positive reaction from the users that were interested to know more this tool. By the results obtained in this evaluation we observed that the implemented features interact well with each other and the users can quickly find what they are looking for.

Author Keywords

Browser; information visualization; revisitation; browsing history.

INTRODUCTION

Nowadays the internet is used worldwide and by a large number of people. A big part of this utilization is spent on browsing websites and the users tend to spend each day more time browsing on the internet. Many of the websites visits are revisitations, they occur when a user returns to page visited previously. A study published by Zhang and Zhao states that 59.6% of the entire web browsing (in browsers with tabs) is intended for revisitation of websites [Zhang11]. The main and current mechanisms of revisitation are the bookmarks, the back button and the history. In the study [Tauscher97] it's mentioned that the back button occupies 30% of all the visits to websites, the bookmarks 3% and the history approximately 1%. However the web browser interfaces have not experienced many improvements in the last years and the revisitation mechanisms have usability problems. The back button works like a stack, when browsing the internet the new

pages are added to the stack, as the old ones are removed, in this way there exists a list with the visited websites of only the current session, not being possible to revisit other previously visited websites [Greenberg99]. The bookmarks throughout the time will have pages that cease to be important for the user and demand a constant management, one task that the users are reluctant to do [Abrams98].

The history, as we saw by the previously mentioned value, is the less used mechanism of the three that were reported and there are some reasons for this fact. The first one is that the history is a "hidden" feature from the main user interface and is not so visible as the back button or address bar. Many users, mainly the beginners, don't even know that a feature like this exists [Weinreich06]. Another very important reason is that the existent history feature of the current web browsers only provides a very rudimentary support to the users. This textual list presents little information related with the websites, a fact that doesn't help the users to find the desired website [Won09][Cockburn99b]. The way as the history systems present the pages (alphabetically or by date) is not the most appropriate way, as the study [Tiernan03] refers. It would make more sense if the history was organized by a topic or idea that provided some sort of context and that it was more alike with the way as the users remember the pages they visited. If the history was organized accordingly with the mental image of the users an improvement in the efficiency of its use would exist.

Another study published by Kandogan and Shneiderman refers that the interfaces of the web browsers are primitive because they don't support many of the user necessities and don't help the users to revisit pages without a large cognitive load [Kandogan97]. These revisitation mechanisms are placed in many different places of the web browser interfaces. If these features were grouped in a central place then the users would have an easier and faster access. Besides revisitations another problem arises with the current web browsers, that is the difficulty of the users to have a context about the current moment of his browsing session. The users many times feel disoriented while exploring pages linked to each other and to remember the path from they had come they need a large cognitive load [Power08].

The visualization nowadays is a very useful process to help to analyze several kinds of information. The human being can acquire more information by the vision than by all the

other senses combined. One of the most important and direct benefits from the data visualization is the large quantity of information that can be quickly interpreted if it is well presented. Other benefits include the possibility of discovering patterns and the interrelationship of information. The visualization can help the strategic reasoning by decreasing the cognitive load and help in the short term memory, allowing to make easier comparisons and inferences [Shneiderman96][Tuftle90][Ware04].

The purpose of our investigation is to create a more effective way of using the browser history using visualization techniques. In particular we want to help the user to be able to manipulate the presented information in the history in an easy way and that could be useful to the user. By this manipulation the user can visualize and analyze the presented information in the way that he desires. Comparing with the traditional history that it's only textual, this system presents many other ways to show the same information to the user and different features. The user can for example filter the history by a specific word and all the results that match with this word are highlighted, or he can verify in which times of the day he visited a specific website.

The system created by us and that we will introduce contains several aspects that help to improve the user browsing experience in the history, as for example the good scalability, the use of visual references about the websites and the possibility of changing the visualization mode of the history. Since usually the user spends a lot of time in his web browsing, the history will also contain a lot of information and a big part of this history belongs to some domains, like for example the domain "Google" that contains several pages like Images, Maps or the Mail. We decided to group all these pages by domain so that the system can have a good scalability.

As we mentioned previously, the visualization of information is very important and a visual reference helps the user to have a better context and a quicker idea about what it represents in comparison with a textual reference. In our work we decided to implement this technique and present the user with a thumbnail of every website when the user is inspecting a specific domain. In this way the user, besides the title and the date, can in a quicker way visualize the structure of the page and remember more easily what was the reason to visit this page in the past.

Still regarding the scalability topic, since the user can have a large time interval in his browsing history, like for example months or years, we implemented a mechanism that allows the user to display the information by day, month or year. In this way all the history is grouped by the period of time chosen by the user.

All these characteristics allow the user to manipulate the shown information in the way that he wants so that he can search in his browsed history in an easier and effective way.

RELATED WORK

In this chapter we will describe three studies related with mainly the visualization of the web browsers history. Other subjects addressed by these projects include features related with the revisitation of websites, what mechanisms exist with the purpose of giving a better browsing context to the users and which are the best ways to present a list of items, in this case the several websites of the web browsing history.

YouPivot

There are several secondary activities that a user can be doing during his web browsing activity, such as listening to music, which can help the user to remember a particular subject that he wants to find. This type of clues can help the user in a special way to find some wanted file for example, this is the goal of the tool YouPivot [Hailpern11].

YouPivot can register several types of activity from the user as for example the visualization of a video or even external activities like reunions. This tool allows also then to relate the time interval in which these activities occurred with other activities that occurred approximately in the same period of time. This logic can be used to correlate with the user browsed history and help to find information in this history.

ActionShot

The following described tool, ActionShot [Li10], pretends to save automatically the web browsing history of the user by interaction objects, such as inserting a value in a field of a form or by clicking in a button. The access to this tool is made by an interface that allows the users an easy control and search. The two bigger goals of the ActionShot are reutilization and content sharing.

This type of saved interactions is a high level of browsing events such as for example the insertion of a new URL in the address bar or the buttons back and advance. The ActionShot contains also a feature that allows to search for interactions executed by the user or for visited websites. The search for user interactions is more efficient than only by website content.

Context Web History

The final described project consists in an extension for a web browser in which the goal is to improve the usability level of the history. One of the main goals of this project is to increase the visibility of the history feature so that it is so visible as the back button or the address bar. Another objective of this project was to improve the search mechanism of the history. This was developed based in a study with users, in which the revisitation of websites was analyzed. The main result of the study was that people can remember more the colors of a website and their structure than only text.

To provide an answer to the characteristics pointed previously, the system Context Web History – CWH [Won09] was created, an extension of the web browser Firefox. This system can be found in a lateral bar of the left side of the browser and provides a feature of searching the user history by a date or keyword. When executing a search in the CWH by a given website name, there is shown the results with a thumbnail associated to each page, the address and the title of the page. When selecting a website from the list it's opened in the main screen on the right place the respective page.

HISTVIS

We will now describe in more detail our system that consists in a mechanism that creates a more effective way of using the web browser history resorting to visualization techniques. Some requirements will be presented next that are fundamental for a good interaction with the user and allows him to browse by the presented information in a simple and effective way. The current history systems don't possess a good search mechanism and presentation of information, relying mainly in showing textual lists of URL's to the user, which is not very pleasant and effective to him. The system that we created intends to improve the usage of the history feature by the user by visualization techniques.

One important point about the user interface is how to show time dependent information. Since that the user history will be receiving more information over the time from the user internet browsing, a way of showing this information accordingly with the elapsed time is necessary. This has to be done in a way that transmits to the user that he is browsing the history throughout the time. The user can have the option of choosing and adjust the time interval that he wants so that he can explore more easily his history.

It has also been previously shown in this document that a visual reference, such as an image, helps in a better way the users to remember a certain subject instead of only textual references. The users associate mainly a website to its colors and visual structure, and therefore it's easier to remember those using thumbnails and icons. Another aspect related with this subject is that usually nowadays each website implements in its structure its own brand image and is known for that image. For example Facebook is known mainly for its blue color and by the timeline that is shown in the middle of the page or YouTube that presents the thumbnails for the several exposed videos.

Another important requirement is how to create a system that has a good scalability, because a problem of the system growing too fast without being prepared might appear. With this fast growth, issues like lack of space or problems in the structure of the system can also appear. To prevent these situations some measures are necessary, the system must be

prepared to receive and show a high number of website visits per day, for example.

A fundamental aspect for our system is the possibility of the user to filter the shown information. As the goal of this system is to show the user web browsing history, there must exist a feature with the purpose of filtering in all shown web domains. After the search string is inserted it's needed to separate the correspondent items from the ones that don't correspond by visualization techniques, with the goal of the user to be able to differentiate easily these items.

It's also necessary a mechanism to collect the user web browsing history in an automatic way. The internet user browsing is recorded automatically and this information must be collected by the system that we implemented.

These are some of the requirements for the correct behavior of our system and we will in the next chapter describe the solutions found by us to correspond to them.

Architecture

The architecture of our system can be divided into two parts, the backend and the user interface. As the names indicate these two parts work in two distinct zones, the backend operates away from the contact with the user and the interface is the connection between the backend and the user by information visualization and input mechanisms.

Given the requirement of collecting the web browsing history automatically and showing it to the user, several possible solutions were thought: Google Chrome, Mozilla Firefox and Internet Explorer. Each one of these options has advantages and disadvantages that we explain in following:

- The Google Chrome has as advantages the fact that it's safe, offers a good support for the programmers and is one of the most used web browsers nowadays. As disadvantages there are the lack of features associated with the browsing history and the usage of a high quantity of RAM memory;
- For the web browser Mozilla Firefox, it has a good safety, offers support to create extensions and can be customized. For the disadvantages it occupies too much RAM memory also and is incompatible with some websites.
- About the Internet Explorer, we have as advantages the fact that it comes already integrated with the operative system Windows and it is compatible with most websites. In the other side this browser is slow compared with other browsers, it doesn't offer a large support to programmers that want to create extensions and contains some security problems.



Figure 1. Initial image of the HistVis

We chose Google Chrome because of its main advantages: it is one of the most used web browsers nowadays and offers a good support to programmers that want to create extensions and applications for this browser. It has a API that allows, in a simple way, to collect information about the user browsed history.

We searched first how was the development of add-ons and what information did Google provide to its programmers. There are three kinds of add-ons that can be developed for Chrome: themes, extensions and applications. The themes are not what we want because they only modify the browser's aspect. Between applications and extensions we decided for an extension because it offers what we want that can be resumed to be a button in the upper right part of the browser's interface and when clicked it opens automatically one tab with the history information.

Interface

About the user interface the most important aspect is the browsed history presentation. This information is generated throughout the time by the user internet browsing. Since it was necessary to show time dependent information we decided to create a vertical timeline that displays the user browsed history throughout the time, this solution is nowadays observed also in other tools such as Facebook, Timeglider (www.timeglider.com) and Twitter. Accordingly with the specified time interval the timeline will be longer or shorter in terms of size. We will in following present a description about the main aspects of HistVis structure and then we will go in more detail in some of these aspects.

General organization

When the user opens the HistVis tool, it's shown the last week of his browsed history. We chose to show this period of time because it is relatively short and recent and so the user can quickly verify any information within the last week. One of the first things that we notice is the quantity of different colors shown in the timeline. Each color represents a particular domain and each domain has always the same color. By this system of colors we can notice if there exists any domain that has been visited more times.

To know more details about the websites included in a particular domain of the timeline we can click in that bar and the sidebar is opened. When a bar is clicked, a red rim is shown in it, by this rim the user knows which bar is the one that he clicked and doesn't lose this context. If the bar where the user clicked was already selected then the rim is removed and the domain in question is no longer selected.

Given that there could exist domains that had a very short number of visits comparing with others from the same temporal instant, that had a very small area to be shown in the timeline or knowing the fact that one of the requirements of this system is that it must be scalable, we decided to group these kind of domains in only one bar. With this grouping action there wouldn't exist a set of domains with such a small area that can't be distinguished from each other. Therefore instead of these ones it's shown a bigger bar with the area of all these domains summed up and the user can verify which are the domains contained in this bar by clicking in it. If the user does this a histogram is shown in the sidebar and the size of each domain depends on the number of visits it had, and they are sorted by this number. In this way the domains that are on top are the ones that had more visits and can be of the interest of the user. On the left side there are the names of the domains and on the right side there is the number of visits.

One of the decided requirements was the existence of a search feature in order for the user to search for a specific domain. This search feature works simultaneously in all timeline, highlighting all the domains that match with the inserted word and blurring the others. When searching for a word, a filter containing this word is added to the upper right place, this helps the user to have a notion of the performed searches and is also possible to remove these filters, which also removes the focus in the timeline to the correspondent bars.

In this search feature there exists an auto-complete mechanism that shows to the user a list of words while he is writing, that match to the written text. All these suggestions had been previously saved while each domain was added to the timeline, all the names of the inserted domains in the timeline are available to be presented to the user as a search suggestion.

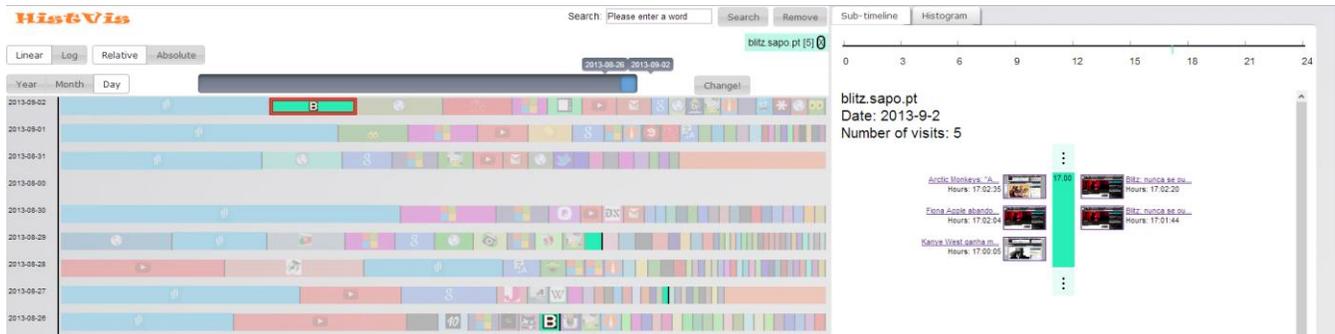


Figure 2. Sidebar with the websites from a domain

In the left upper side of this system we can visualize several buttons, among them there are three that allow the user to change the type of temporal visualization of the timeline. It's possible to choose the mode of day, month or year. The day mode was already presented and it's the default time mode shown to the user. The month and year mode group all the domains by these time instants and allow to show to the user a more compact set of information. The advantages of these modes are that the HistVis has a way of scaling well the presented information and the user has the chance of knowing quickly which is the domain that was more visited in a month or year. Regarding the domains that have a very low size to be shown in the timeline, the result is the already explained previously, they are grouped and then is possible to open this list on the sidebar.

In the temporal mode of year the selected history is grouped by year which means that there is a lot of information in a small area. This can be especially useful for the user in case he wants to do a list of the most visited domains of his browsed history. To be possible for the user to verify immediately which day, month or year has more visits without having to count this number manually domain by domain, we decided to implement the possibility of changing the presentation mode of the user's history. It's possible to change the size of the bars accordingly with the total number of visits. When this feature is used there is a bar that contains the largest number of visits of the shown history and therefore this is the bigger bar that ends in the right limit of the timeline, the others have their size calculated accordingly with the number of visits in a percentage format comparing to the bigger bar. The advantage of this aspect is that the user knows immediately which is the time instant that had more or less visits accordingly with the size of the bars. It's at the same time a timeline of the user browsed history but also a vertical bar graph.

Another aspect similar with the previously explained one is the logarithmic visualization mode that allows changing the size of the bars of each domain in a linear perspective to a logarithmic one. This option can be especially useful for combining with the absolute visualization mode because in case there exists a large difference of visits between the

websites of the different days, months or years, with this option this large difference doesn't exist anymore and it turns less notorious. In a specific case we can suppose that in two days, one of those says contains 100 visits in total and the other one only 5 visits. As we can imagine the difference will be very big having a bar with a large size and the other one with a reduced size. With this visualization mode this difference will be more faded and the size of each bar will be closer from each other.

Backend

Regarding the operations made in background the main idea is the collection of the user browsed history data by the Google Chrome API. One API indicates how some software elements should communicate between themselves. By the methods provided by a system we can interact with internal functions from that system and collect information. For this we used the API for the history, which allows to add, remove and query visited pages of the web browser history. There are several methods available in this API, which we list some:

- Search;
- GetVisits;
- AddURL;
- DeleteURL;
- DeleteRange;
- DeleteAll.

Accordingly with the provided information about each method we concluded that the most adequate method for our system was the Search one. This method allows searching for visited websites which relate to one particular string, although we can leave this field without any value so that it displays all the available results. By this method we can obtain the following data:

- Identifier of each item;
- Address of the visited website from the user;
- Title of the visited page;

- When was the page visited for the last time;
- The number of visits of the page done by the user;
- The number of visits of the page done by the user by manually inserting the address in the address bar.

These data after retrieved by the Chrome's API are displayed to the user by a multidimensional array, of which in each position contains the information described previously for each visited website. Later we can collect these information from the array by a cycle.

The first operation that we executed was to sort all the collected data from the oldest to the most recent date, in this way we were sure that the data were correctly sorted in order to show them also in this way to the user in the timeline. Since the data that show the time of which the page was visited for the last time are in milliseconds we had to create an instance of Date and collect the complete date with methods provided by the standard library of Javascript for data of type Date. Following with the date of each visited website already converted to an accessible format is saved in an array together with the remaining information collected from the API, with the title and URL.

Next we removed the duplicated dates from the visited websites because the goal of our timeline was to display the browsed history by day, month or year, and show these dates in the left column of the timeline. We decided to show the dates in this way in order to group the user history by temporal groups and put the system more scalable.

The following action was to fill the right column with the respective websites. As already previously mentioned all the visited websites from the same domain are grouped in only one bar and all these bars are sorted by the largest number of visits to the least one. For this operation we used a multidimensional array of which for each position we saved the name of the domain and its number of visits. We created two cycles that iterated all the saved data by the Chrome's API and the array that we use to save the counting of visits of each domain. We verified if the observed domain was already contained in this array, if it was then we added the number of visits in one number. If after iterating all the positions of the array the domain wasn't saved inside it, we created a new position with the method Push, with the new domain name and one visit.

In this chapter we analyzed the main points of our system. These main characteristics are the websites grouped by domain, the possibility of opening these websites in the sidebar and analyze with more detail the time that they were visited, the possibility of searching for a particular word and highlight the matching results in the timeline, and lastly the different visualization modes of the browsed history. With all these features HistVis offers a large variety of visualization techniques with the goal of helping the user to find the desired information in his browsed history.

EVALUATION

We will now present the evaluation of our system by tests with users. These tests had the goal of evaluating the quality of the work by the point of view of the end user of the system. The end user has a different perception than the product creator because he doesn't have the same pre-conceived ideas about which are the best ways of developing and implementing the product features. The purpose of these tests is to observe and collect data from the utilization of the system by the users in order to find possible errors and improve on certain areas.

The tests were made with several users and this gave the possibility of having a list with some points to improve, being these ones in a certain way sorted by the level of importance since the most relevant points were identified by more users. We made tests with 20 users, this is number is large enough to know many aspect that can be improved. These tests with the users allowed observing for example which features the users had more difficulties.

Methodology

The main idea of this evaluation was to gather a set of people, explain them in a simple way the mechanism of this system and next do some tests by the execution of several tasks. Each one of these tasks involves the utilization of several features of the system and allows to observe in which way these features can help the user to do the tasks. It was used always the same test environment with the same history data.

Initially a simple explanation was given to the users about what was the idea of this system and what could it offer. The users could also explore freely the system by some minutes in order to gain a basic knowledge of the system and its features. Following this a list with the tasks was given to the users and each one of these tasks already had a pre-defined result in order to know if the users had executed correctly the tasks or not. We observed the users to execute the tasks and their reactions. We also wrote their main difficulties and errors. We tried to interact the less possible with the users during the execution of the tasks in order to not contaminate the results. Finally the order of the tasks was alternating for each user so that the first task wasn't always the same.

Besides the results from these tests we asked the users to fill a survey in the end of the tests. This survey has the name System Usability Scale (SUS) [Brooke96] and it is composed by 10 questions. This scale allows to help to answer the question "Is the system A more usable than the system B?" because it provides a high level view of usability and is usually used for usability comparisons between systems. This survey has as a result a value between 0 and 100 that represents a measure of the general usability level of the analyzed system. To calculate this result we have to use the following procedure. The value of each item varies from 0 to 4; for the values 1, 3, 5, 7 and 9

the value is obtained by subtracting the assigned position by the user by one. For items 2, 4, 6, 8 and 10 the value is obtained by subtracting 5 to the position assigned by the user. In the end we multiply the sum of all these values by 2.5 to obtain the total value of the SUS.

The participants of the tests accepted to enter in this evaluation by free will after a request personally by us. The most of the tests was executed in the college Instituto Superior Técnico since there was a big possibility of finding there voluntaries. The test environment consisted in the same computer for every test and with the same data previously collected and organized, in the web browser Google Chrome and an opened tab with the initial screen from HistVis. In general the evaluation with each user took an average of 15 minutes.

Users

All the users accepted by free will to participate in the proposed tests. These users have an age between 18 and 29 years and are in a certain way familiarized with the use of technologies, and so they are used to explore applications similar with the one that we created. None of the users had already used our system previously. From the twenty users, nineteen are men and one is a feminine volunteer. Seventeen users possessed also superior education and all of them spent many time browsing the internet.

Tasks and features

In order to test the several features of the system, it was asked to the users to execute several tasks related with the browsed history manipulation. These tasks aren't executed always in the same order so that the first one is not always that one where the users have more difficulties since they haven't experience with the system. The tasks that were provided to the users were:

1. See what is the day of the last week in which there exists the largest number of visits;
2. Discover in how many days, in the past 3 weeks, the domain "Sapo" was visited. Also verify if any of those days the domain "Sapo" was the most visited one (If yes, which day?);
3. See in the past 2 months what is the domain that was visited more times;
4. See in yesterday day, in the most visited domain, what pages were visited from that domain and which times of the day were they visited;
5. See in yesterday day which periods of the day the user was more active;
6. See in the past 3 weeks which were the visited websites that belong to the domain "Sapo", for instance "Blitz", "Visão", "Auto Sapo" and "Sapo Emprego".

One of the purposes of the execution of these tasks was to analyze the existent features by the users and find out possible improvements that could be implemented. Each task included the execution of at least one feature to complete it. The main tested features were the following ones:

- Absolute visualization mode;
- Changing of the time interval;
- Filtering;
- Scale type;
- Sidebar.

All these features were useful for the user in the execution of the tasks of this evaluation. It was pretended that almost any task repeated the already executed features and that we didn't have a very extent list of tasks because it could occupy too much time to the users.

For the first task, it was reasonable simple because when the application is loaded to user's screen it is shown right away the last week of the browsed history, and by this way the user only needed to search in this set of information the pretended objective. In this case was necessary to know the day that had the biggest number of visits and for this we had two choices: or we counted manually the visits of each domain or we used the absolute visualization mode in order to change the size of the bars of each day accordingly with the number of visits.

On task number 2 it was necessary to use other features such as changing the time interval of the user's history with the help of the bar positioned on the top area. So the first step that the user had to do was to change the interval of the history to the past 3 weeks and in the set of information search for how many days there have been visits to the domain www.sapo.pt. Once again the users could verify this information manually, analyzing all domains from each day and see if they belonged to Sapo's domain. However there exists another easier alternative for the user that is to search for the string "www.sapo.pt" in the search field. The next requested action was to verify if each one of these domains found by the user was the one that contained more visits. To verify this this situation the user could check if the specified domain was the one in the left side, and if this situation was true then it was confirmed that it was the most visited domain from that day.

In the following task, the third one, once more the user had to use the change of the time interval feature but in this case for the past 2 months. In next it was necessary to verify what was the domain that had the largest number of visits in those 2 months and in this action the users could wrongly state that this domain was the bar with a had a bigger area but this wasn't a sure correct answer because there could exist other domains not with a bar so big but with more occurrences during these 2 months, and so with the sum of

User/Tasks	1	2	3	4	5	6
A	15	20	18	17	21	12
B	10	16	15	13	18	10
C	17	21	20	18	23	13
D	20	24	19	20	25	12
E	8	13	13	14	19	9
F	16	20	19	17	23	10
G	12	14	11	13	16	14
H	10	17	14	16	20	9
I	15	19	23	18	21	9
J	17	23	22	19	21	12
K	17	22	25	19	24	20
L	20	23	27	25	26	23
M	15	20	26	17	24	18
N	12	17	23	15	24	16
O	15	19	26	24	22	22
P	17	24	25	18	21	19
Q	17	21	26	21	25	16
R	15	25	23	27	25	23
S	16	27	28	24	22	20
T	19	24	25	22	27	24
Maximum	20	27	28	27	27	24
Minimum	8	13	11	13	16	9
Average	15,15	20,45	21,4	18,85	22,35	15,55
Stand. Deviation	3,1980	3,6259	4,9234	3,8765	2,7253	5,0938

Table 1. Time took to execute the tasks (in seconds)

the total visits it would be larger. One way to do this verification was to change the visualization mode from day to month or year, and in that case we could see the domain that was in the left of the timeline.

In the following task the user had to look for yesterday's day, check what was the domain with more visits and analyze its details. The user could simply look for the second line and search for this information. Then the user could click in the domain with more visits (the left one) and the pages contained in this domain were opened in the sidebar. In the sidebar the user can verify in what times of the day there were occurred visits to this domain.

On the fifth task the procedure was very similar with the previous task but here what was wanted was to use another feature from the sidebar that is the horizontal sub-timeline, and besides this, the user had to select several domains from the same day at the same time. In this horizontal sub-timeline it's possible to visualize in what times of the day the user visited websites by looking at the exposed traits in the correspondent place in its hour. Since what was asked was to verify in which periods of the day the user was more active, the goal was to select all domains of the day and verify this information in the horizontal sub-timeline.

Finally for the last task was necessary to change the time interval for the past 3 weeks and search for the word “Sapo” in order to display all the websites that were related with that word.

Results

Accordingly with the obtained results we will now look to all the data together and verify if the tested features were efficient for the users or not. By the took time, number of errors and reaction from the users we will have an idea about if the existent features can be improved and find out if the addition of other mechanisms can facilitate more the usability level with the user.

The using of the features in a general way was well accepted by the users and helped them to reach the goals of the tasks faster. In a concrete case, in the first task, the objective was to verify which was the day of the last week that had more visits in total. The user could simply count the number of visits of each domain separately and reach to the requested result but this wasn't a very efficient way, since it was possible to change the visualization mode to the absolute one and reach to the goal in less time. The user E, for example, as soon as he realized what was the pretended objective user immediately this feature and reach the result in only 8 seconds, as the user L took 20 seconds since he started to count the number of visits individually and only then changed the visualization mode to the absolute one.

The changing of the time interval is one of the main features of our system and had to be used in three of the tested tasks. In this feature there was some difficulties in manipulating the bar to choose the pretended initial date because to select one specific day it was necessary a lot of precision and sometimes the users passed through the date more than once to finally reach the correct target. However the users understood quickly what was the goal of this bar and as soon as they realized that they had to change the time interval in some task the users found this feature faster than for example the buttons to change the visualization mode of the timeline. Another difficulty related with the change of the time interval was to calculate which was the day to choose, for example in the second task it was necessary to select the last three weeks but the users had to check the calendar or took some time to find the correct date, this resulted in an increase of the time to perform the tasks related with this feature.

As in the majority of the websites there exist a search feature, the users quickly understood what was the goal of this field, available on our system in the right upper part. When the user was asked to analyze a particular website in the tasks, the users quickly resorted to this feature. With the automatic suggestions the users decreased in some time the execution of the tasks. With the implementation of visualization techniques related with the filtering, such as the reduction of the opacity of the items that didn't correspond with the search string, the task of the users was

facilitated and they could see quickly the bars that they were looking for. In the second task the users could verify immediately if there was any domain related with “Sapo” that was the most visited of the day because the user only had to look for a bar with a normal level of opacity in the left side of the timeline. In terms of time this feature was very intuitive to the users and didn't occupy much time using it.

In the third task it was fundamental that the type of scale of the user history was changed to month or year. In this task there existed many users choosing the wrong answer. The reason for this low level of success is because what was requested was to verify the domain in the past 2 months with the largest number of visits and the users wrongly searched for the bar with the largest area and assumed that this was the domain with more visits. In this task there exists a very large difference between the minimum and maximum value, respectively 11 and 28 seconds. The user G for example changed the type of scale right away to year and was able to see quickly what was the domain with more visits in the two selected months, lasting 11 seconds and not having any error.

The implementation of the sidebar was able to increase the level of scalability of the system and show together with some details the websites grouped in the domains in the main timeline. In the execution of the tasks some users didn't had understood what was the goal of this tab and its utility. When was asked to check the details of some domains there was some slowness from the users in clicking in the bars, because they had not figured that this was the way to open the sidebar and see the respective details. However with the sidebar already opened the sub-timeline was well understood and the users quickly knew how the structure in the area was positioned. Especially for the task 5 there existed some difficulties in selecting several domains from the same day at the same time as it was pretended. The users were unaware that it was possible to do this for later visualizing the hours in the horizontal sub-timeline and so they took a lot of time in this situation. This task was the one with a larger number of seconds to be terminated.

In general the users realized the tests with a reasonable performance, the disposition of the browsed history by a timeline was intuitive and the users didn't had much difficulties to understand this mechanism.

CONCLUSION

Despite the revisitation of websites being included in a large part of the entire web browsing, the current web browsers have several usability problems in their revisitation mechanisms. Of these mechanisms, the history is one of the less used by the users. Some reasons for this are because of the fact that this mechanism is usually far from the main interface of the web browsers and also because of the difficulty of the users to use this textual list

to find the desired website. Another problem is the difficulty of the users in remembering in which part of their browsing session they are in the current moment. To know more about these problems and possible solutions, some existent projects about this topic were gathered and explained. The main themes of these projects are the web browser history visualization, the revisitation of websites and display of information.

The more relevant projects for the pretended goals were [HailPern11], [Li10], [Won09], [Zhou08] and [Shirai06]. These were the most relevant because they are the ones that have most resemblances with the main goal of this system that is the web browser history visualization. However none of these projects really solves the problem, since none has in account the use of additional information to contextualize the user, the use of thumbnails and the use of scalability techniques.

One of the formulated requirements was to show time dependent information. Since the user browsed history has dates related with all the occurrences, we needed also a way to show this information throughout the time. After an investigation we concluded that the best way to show this type of information was to exist a surface that slides in the vertical or horizontal and the area in which this surface slides represents the elapsed time.

A visual reference is an important aspect in the visualization of information, by this reference a user can remember in an easier way and relate faster a particular subject. The users associate mainly a website to its colors and visual structure, comparing with for example only a textual reference such as the title of the website. In our system we used thumbnails and colors to represent the websites and domains. The structure of each website in the sidebar is shown by a thumbnail that can be zoomed by putting the mouse pointer on top of them and the color of each bar in the timeline is generated accordingly with the name of its domain.

All these characteristics help the user to find the information that he is looking for in the browsed history in an easy and effective way. By the tests realized with users we verified that the implemented features made possible to reach the pretended goal of each task in a faster way. By the gathered comments we observed that this system is interesting and provided a pleasant experience for the users.

For the future work we would like to improve the main aspects pointed during the tests with users, like for example a better mechanism of selection of the time interval or to implement some ways to help the user to contextualize better with the existent features. Another important issue that we would like to add to our system is the implementation of more than one timeline containing contextual information, such as for example the applications opened in the foreground in the user's computer or the events registered in the personal calendar

of the user. Finally we would like to improve the performance of this tool by using a database and cache memory to store the retrieved data from the user browsed history.

REFERENCES

- [Brooke96] Brooke, J. SUS: a "quick and dirty" usability scale, in P. W. Jordan, B. Thomas, B. A. Weerdmeester, & A. L. McClelland. *Usability Evaluation in Industry*. 1996. Pp. 189-194, London: Taylor and Francis.
- [Hailpern11] Hailpern, J., Jitkoff, N., Warr, A., Karahalios, K., Sesek, R., Shkrob, N. YouPivot: improving recall with contextual search, in *Proceedings of the 2011 annual conference on Human factors in computing systems*. ACM, 2011. Pp. 1521-1530, Vancouver, Canadá. ACM press.
- [Li10] Li, I., Nichols, J., Lau, T. Here's what i did: sharing and reusing web activity with ActionShot, in *Proceedings of the 28th international conference on Human factors in computing systems*. ACM, 2010. Pp. 723-732, Atlanta, EUA. ACM press.
- [Shirai06] Shirai, Y., Nakakoji, K., Yamamoto, Y. A history-centric approach for enhancing web browser experiences, in *CHI '06 extended abstracts on Human factors in computing*. ACM, 2006. Pp. 1319-1324, Montreal, Canadá. ACM press.
- [Won09] Won, S.S., Jin, J., Hong, J.I. Contextual Web History: using visual and contextual cues to improve web browser history, in *Proceedings of the 27th international conference on Human factors in computing systems*. 2009. Pp. 1457-1466, Boston, EUA. ACM press.
- [Zhang11] Zhang, H., Zhao, S. Measuring web page revisitation in tabbed browsing, in *Proceedings of the 2011 annual conference on Human factors in computing systems*. ACM, 2011. Pp. 1831-1834, Vancouver, Canadá. ACM press.
- [Zhou08] Zhou, Wei. Lifestream – Redesign history and bookmarking, [online], [publicado 2008-07-07], [referenciado 2011-09-22]. Disponível na internet: <URL: <http://weizhou.wordpress.com/2008/07/07/prototype/>>.